



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/864,608	05/23/2001	Peter J. Brittenham	RSW920010107US1	3650
25259 7590 10/05/2011				
RSW IP Law IBM CORPORATION 3039 CORNWALLIS RD. DEPT. T81 / B503, PO BOX 12195 RESEARCH TRIANGLE PARK, NC 27709				
EXAMINER				
MEKY, MOUSTAFA M				
ART UNIT		PAPER NUMBER		
2457				
NOTIFICATION DATE		DELIVERY MODE		
10/05/2011		ELECTRONIC		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

RSWIPLAW@us.ibm.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

*Ex parte* PETER J. BRITTENHAM, DOUGLAS B. DAVIS, DAVID B.  
LINDQUIST, and AJAMU A. WESLEY

---

Appeal 2009-010372  
Application 09/864,608<sup>1</sup>  
Technology Center 2400

---

Before JEAN R. HOMERE, THU A. DANG, and MICHAEL R. ZECHER,  
*Administrative Patent Judges.*

ZECHER, *Administrative Patent Judge.*

DECISION ON APPEAL

---

<sup>1</sup> Filed on May 23, 2001. The real party in interest is International Business Machines Corp. (App. Br. 2.)

## I. STATEMENT OF THE CASE

Appellants appeal under 35 U.S.C. § 134(a) (2002) from the Examiner's final rejection of claims 1-29. (App. Br. 2.)<sup>2</sup> We have jurisdiction under 35 U.S.C. § 6(b) (2008).

We affirm.

### *Appellants' Invention*

Appellants invented a method, system, and computer program product for dynamically updating or redeploying services, such as web services or other network-accessible services, at various sites within a network. (Spec. 2, ll. 2-4.)

### *Illustrative Claim*

1. A method of dynamically redeploying web services in a computing network, the method comprising:

receiving a redeployment trigger for a selected web service wherein the selected web service includes executable code;

determining one or more network locations where the selected web service including the executable code has been deployed from its original location at an origin server;

programmatically removing the selected web service including the executable code from the network locations and the origin server; and

programmatically replacing the selected web service at the network locations and the origin server.

---

<sup>2</sup> All references to the Appeal Brief are to the Supplemental Appeal Brief filed on January 16, 2008, which replaced the Appeal Brief filed on February 20, 2007.

*Prior Art Relied Upon*

Dugan	US 6,363,411 B1	Mar. 26, 2002 (filed on Oct. 19, 1999)
Onyeabor	US 6,631,512 B1	Oct. 7, 2003 (filed on Jan. 15, 1999)
Robotham	US 6,704,024 B2	Mar. 9, 2004 (filed on Nov. 29, 2000)

*Rejections on Appeal*

Claims 1-11 and 14-29 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the combination of Dugan and Onyeabor.

Claims 12 and 13 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the combination of Dugan, Onyeabor, and Robotham.

*Appellants' Contentions*

Appellants contend that since Dugan's disclosure relates to call processing services as opposed to web services, Dugan does not teach or suggests "programmatically removing [a] selected web service including executable code," "programmatically replacing the selected web service," and "receiving a redeployment trigger," as recited in independent claim 1. (App. Br. 9-11.) Moreover, Appellants argue that Onyeabor's disclosure of a web page that includes executable code fails to remedy the alleged shortcomings in Dugan. (*Id.* at ll. 11-12.) Further, Appellants allege that the Examiner does not provide a sufficient rationale to combine Onyeabor's web page development, deployment, and execution with Dugan's telecommunication service processing. (*Id.* at 12-13; Reply Br. 4-5.)

*Examiner's Findings and Conclusions*

The Examiner finds that Dugan's disclosure of removing and replacing a selected service, in conjunction with Onyeabor's disclosure of a

web page document that includes executable code, teaches or suggests “programmatically removing [a] selected web service including executable code,” and “programmatically replacing the selected web service,” as claimed. (Ans. 12-13.) Moreover, the Examiner finds that Dugan’s disclosure of replacing a service with a new version of the service requires a trigger for replacement or redeployment to work and, therefore, teaches or suggests “receiving a redeployment trigger,” as claimed. (*Id.* at 13-14.) Further, the Examiner provides a motivation that would have led an ordinarily skilled artisan to combine Dugan and Onyeabor. (*Id.* at 14.) The Examiner asserts that such motivation constitutes a sufficient rationale for the proffered combination. (*Id.*)

## II. ISSUE

Have Appellants shown that the Examiner erred in concluding that the combination of Dugan and Onyeabor renders independent claim 1 unpatentable? In particular, the issue turns on whether:

- (a) the proffered combination teaches or suggests “receiving a redeployment trigger,” as recited in independent claim 1;
- (b) the proffered combination teaches or suggests “programmatically removing the selected web service including the executable code,” as recited in independent claim 1;
- (c) the proffered combination teaches or suggests “programmatically replacing the selected web service,” as recited in independent claim 1; and
- (d) the Examiner provides a sufficient rationale for the proffered combination.

### III. FINDINGS OF FACT

The following Findings of Fact (hereinafter “FF”) are shown by a preponderance of the evidence.

#### *Dugan*

FF 1. Dugan’s figure 5(c) depicts a Service Administration (hereinafter “SA”) component (500) that includes an Inventory Manager (516), a Database of Record (hereinafter “DBOR”) Manager (520), an Environment Manager (530), an Audit and Reconciliation Manager (535), and a Monitoring and Logging Manager (540). (Col. 15, ll. 56-61.) In particular, Dugan discloses that the Inventory Manager (516) or DBOR Manager (520) notifies the Environment Manager (530) whenever the DBOR is modified, such as when a service has been replaced with a new version. (Col. 17, ll. 6-9.) Dugan discloses that the Environment Manager (530) ensures that each service node that is impacted receives the update (i.e., receives the new version of the service). (*Id.* at ll. 9-12.)

FF 2. Dugan discloses inputting and storing both the service node and service profiles in an SA to enable automatic tracking of: 1) the capabilities and capacity of each service node; 2) the services and data to be deployed to each service node, and when; and 3) the configuration of service execution. (Col. 20, ll. 14-21.) Dugan discloses maintaining the capabilities of each node and computer in the network so that simple and complex business rules governing data or service distribution, activation, and removal may be applied to optimize the execution of services on the service nodes. (*Id.* at ll. 22-26.)

*Onyeabor*

FF 3. Onyeabor discloses a web page development tool which enables a developer to create a web page document that includes executable code, thus eliminating the need to download foreign executables during the display and manipulation of the web page. (Col. 6, ll. 13-18.) According to Onyeabor, this eliminates the risk that malicious code will be downloaded and allowed to wreak havoc on a client machine. (*Id.* at ll. 18-20.)

FF 4. Onyeabor discloses that the deployment of a web page refers to the act of sending the web page to a server, which in turn provides client computers access to the web page via the Internet. (Col. 16, ll. 38-41.)

IV. ANALYSIS

*35 U.S.C. §103(a) Rejection—Combination of Dugan and Onyeabor  
Claims 1, 18, and 19*

Independent claim 1 recites, *inter alia*, 1) “receiving a redeployment trigger;” 2) “programmatically removing the selected web service including the executable code;” and 3) “programmatically replacing the selected web service.”

As detailed in the Findings of Fact section *supra*, Dugan discloses an SA component that includes an Inventory and DBOR manager, both of which are capable of notifying an Environment Manager that a service has been updated. (FF 1.) Dugan discloses that the Environment Manager ensures that each node impacted by the update receives the new version of the service. (*Id.*) In particular, we find that Dugan’s disclosure of the Environment Manager receiving notification from either the Inventory or DBOR manager to deploy a new version of a service to each node impacted

by the update teaches or suggests receiving a trigger that indicates the deployment of a new version of a service to each node in a computer network where the service was originally deployed. Thus, we find that Dugan teaches or suggests “receiving a redeployment trigger,” as recited independent claim 1.

Further, Dugan discloses that the SA component stores the capabilities of each node so that rules governing service distribution, activation, and removal may be applied to optimize the execution of services on each node. (FF 2.) In particular, we find that Dugan’s disclosure of removing a service from each node, in conjunction with deploying a new version of the service to each node (FF 1), teaches or suggests removing and replacing services on each node located in a computing network.

Next, Onyeabor discloses a web page development tool which enables a developer to create a web page that includes executable code. (FF 3.) Onyeabor also discloses deploying the web page to a server that provides client computers access to the web page via the Internet. (FF 4.) In particular, we find that an ordinarily skilled artisan would have understood that Onyeabor’s disclosure of deploying a web page that includes executable code amounts to deploying a web page that provides web services via the executable code included therein. In summary, we find that ordinarily skilled artisan would have readily appreciated both removing and replacing Onyeabor’s selected web page that includes executable code from or on each node located in Dugan’s network where the selected web page was originally deployed. Thus, we find that the combination of Dugan and Onyeabor teaches or suggests “programmatically removing the selected web



service including the executable code,” and “programmatically replacing the selected web service,” as recited in independent claim 1.

*Rationale to Combine*

We are not persuaded by Appellants’ argument that the Examiner does not provide a sufficient rationale for the proffered combination. (App. Br. 12-13; Reply Br. 4-5.) The U.S. Supreme Court has held that “[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results.” *KSR Int’l Co. v. Teleflex, Inc.*, 550 U.S. 398, 416 (2007). The Court further instructs that:

[o]ften, it will be necessary for a court to look to interrelated teachings of multiple patents; . . . and the background knowledge possessed by a person having ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in a the fashion claimed by the patent at issue.

*Id.* at 418. Additionally, the Court instructs that:

“[[r]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.”... However, the analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ.

*Id.* (citation omitted).

Upon reviewing the record before us, we find that the Examiner’s suggestion for the proposed modification in the prior art suffices as an articulated reason to establish the *prima facie* case of obviousness. In summary, we find that an ordinarily skilled artisan at the time of the claimed

invention would have combined Dugan's disclosure of a trigger that indicates deploying a new version of a service to each node where the service was originally deployed, removing the service from each node, and replacing the service on each node with the new version of the service (FFs 1 and 2), with Onyeabor's disclosure of deploying a web page that includes executable code. (FFs 3 and 4.) This proffered combination would predictably result in an efficient way to deploy or update a web page that provides web services via the executable code included therein. (FF 4.) Moreover, deploying or updating a web page that includes executable code eliminates the risk of downloading malicious code since there is no need for a client computer to download foreign executables during the display and manipulation of such web page. (FF 3.)

Further, as prescribed by the controlling case law, while it is often necessary for an Examiner to identify a reason for combining the familiar elements obtained from the prior art in establishing a *prima facie* case of obviousness, the identification of such a reason is not a *sine qua non* requirement. *See KSR* 550 U.S. at 418-19. So long as the Examiner provides an articulated reasoning with some kind of a rational underpinning to substantiate the obviousness rejection, such a conclusion is proper. *See id.* at 418. In this case, the Examiner provided more than just a mere conclusory statement. The Examiner states that it would have been obvious to an ordinarily skilled artisan at the time of the claimed invention to modify Dugan in view of Onyeabor to use a selected web service wherein the selected web service includes executable code. (Ans. 4 and 14.) According to the Examiner, an ordinarily skilled artisan would be motivated to do so because the deployment or redeployment of web pages is an efficient way of

sending and updating web pages, and using a web page that includes executable code virtually eliminates the risk that malicious code will be downloaded. (*Id.*) In our view, such statements suffice as an articulated reason with a rational underpinning to support the proffered combination.

Independent claims 18 and 19 either recite the same or similar claim limitations as independent claim 1. (*See* App. Br. 12-13; *see also* Claims App'x.) Accordingly, for the same reasons discussed *supra*, the Examiner has not erred in concluding that the combination of Dugan and Onyeabor renders independent claims 1, 18, and 19 unpatentable.

*Claims 2, 20, and 21*

Appellants contend that Dugan's SA component, which maintains the services or data deployed to each service node, fails to teach or suggest that "the redeployment trigger compromises a redeployment request from the origin server," as recited in dependent claims 2, 20, and 21. (App. Br. 13-14.) We do not agree.

As discussed *supra*, since Dugan discloses that the trigger is initiated by either the Inventory or DBOR manager located within the SA component (FF 1), we find that an ordinarily skilled artisan would have understood that the trigger includes a deployment request from the SA component itself. Thus, we find that Dugan teaches or suggests the disputed claim limitation. It follows that the Examiner has not erred in concluding that the combination of Dugan and Onyeabor renders dependent claims 2, 20, and 21 unpatentable.

*Claims 3, 22, and 23*

Appellants contend that while Dugan discloses business rules that govern data or service distribution, activation, and removal, Dugan fails to

teach or suggest “sending the redeployment trigger when the selected web service including the executable code is to be revised,” as recited in dependent 3, and similarly recited in dependent claims 22 and 23. (App. Br. 15-16.) We do not agree.

In light of our analysis *supra*, we find that an ordinarily skilled artisan would have appreciated sending Dugan’s trigger when Onyeabor’s selected web page that includes executable code needs to be updated or revised with a new version of the selected web page. Thus, we find that the combination of Dugan and Onyeabor teaches or suggests the disputed claim limitation. It follows that the Examiner has not erred in concluding that the combination of Dugan and Onyeabor render dependent claims 3, 22, and 23 unpatentable.

*Claims 4-11 and 14-17*

Appellants do not provide separate and distinct arguments for patentability with respect to dependent claims 4-11 and 14-17. Therefore, we select independent claim 1 as representative of these cited claims. Consequently, the Examiner has not erred in rejecting dependent claims 4-11 and 14-17 for the same reasons set forth in our discussion of independent claim 1. *See* 37 C.F.R. § 41.37(c)(1)(vii).

*Claims 24, 26, and 28*

Appellants contend that Dugan’s disclosure of data or service removal in a telecommunication network fails to teach or suggest “removing the selected web service from all of the network location[s] where the web service has been deployed,” and “replacing the selected web services at all the network locations where the web service has been deployed,” as recited in dependent claims 24 and 26, and similarly recited in dependent claim 28. (App. Br. 16-17.) We do not agree.

As discussed *supra*, we find that ordinarily skilled artisan would have readily appreciated both removing and replacing Onyeabor's selected web page that includes executable code from or on each node located in Dugan's network where the selected web page was originally deployed. Thus, we find that the combination of Dugan and Onyeabor teaches or suggests the disputed claim limitations. It follows that the Examiner has not erred in concluding that the combination of Dugan and Onyeabor render dependent claims 24, 26, and 28 unpatentable.

*Claims 25, 27, and 29*

Appellants contend that while Dugan discloses provisioning based on load balancing, efficiencies, and demand, Dugan fails to teach or suggest "replacing the selected web service at the network locations comprises replacing the selected web service with an updated web service including updated executable code," as recited in dependent claim 25, and similarly recited in dependent claims 27 and 29. (App. Br. 17-18.) We do not agree.

As discussed *supra*, since Dugan discloses deploying a new version of a service to each node impacted by an update (FF 1), we find that an ordinarily skilled artisan would have understood replacing Onyeabor's selected web page that includes executable code with a new or updated version of the selected web page that includes new or updated executable code. Thus, we find that the combination of Dugan and Onyeabor teaches or suggests the disputed claim limitation. It follows that the Examiner has not erred in concluding that the combination of Dugan and Onyeabor render dependent claims 25, 27, and 29 unpatentable.

*35 U.S.C. §103(a) Rejection—Combination of Dugan, Onyeabor, and Robotham*

*Claims 12 and 13*

Appellants do not provide separate and distinct arguments for patentability with respect to dependent claims 12 and 13. Therefore, we select independent claim 1 as representative of these cited claims. Consequently, the Examiner has not erred in rejecting dependent claims 12 and 13 for the same reasons set forth in our discussion of independent claim 1. *See* 37 C.F.R. § 41.37(c)(1)(vii).

V. CONCLUSIONS OF LAW

The Examiner has not erred in rejecting claims 1-29 as being unpatentable under 35 U.S.C. § 103(a).

VI. DECISION

We affirm the Examiner's decision to reject claims 1-29 as being unpatentable under 35 U.S.C. § 103(a).

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

ELD